

Student Name:

Student id:

Sect #: Serial #:

University of Bahrain

College of Information Technology  
Department of Computer Science

**ITCS332: Concepts of Programming Languages**

**Quiz #1: Chapter 1**

\*\*\*\*\*

**True/False Question:**

- 1)   T   C++ programs are faster in execution than equivalent JAVA programs.
- 2)   T   Most scripting languages are implemented using interpreting systems.
- 3)   F   A language based on inheritance and abstract data types is imperative.
- 4)   F   Increasing the feature multiplicity improves the language readability.
- 5)   F   Built-in operator overloading is good for readability of programs.

**Fill in blanks Question:**

- 1) Give 2 issues (factors) that reduce the overall simplicity of a programming language :
  - A language allows much feature multiplicity.
  - A language includes a large number of constructs.
  - A language allows much operator overloading.
- 2) Suggest 2 language features to reduce the execution time of programs in a new language:
  - No checking of array indices against bounds.
  - Perform data type checking during compilation time.
  - Include optimization step in the compiler.
  - Use preprocessor macros rather than subprograms (Functions).
- 3) The Semantics analyzer accepts as input a **parse tree** and generates as output an **intermediate code**.
- 4) The main bottleneck in interpreting systems is **Statement decoding**.

Student Name:

Student id:

Sect #: Serial #:

University of Bahrain

College of Information Technology  
Department of Computer Science

**ITCS332: Concepts of Programming Languages**

**Quiz #1: Chapter 1**

\*\*\*\*\*

**Fill in blanks Question:**

- 1) Suggest 2 language features to improve the C++ reliability:
  - Exclude pointers from C++.
  - Perform checking of data types at compilation time.
  - Include checking of array indices against bounds.
- 2) Suggest 2 language features to reduce the compilation time of programs in a new language:
  - Exclude preprocessor macros.
  - Reduce the degree of optimization (No optimization).
- 3) A code generator accepts as input **an intermediate code** and generates as output **a machine code**.
- 4) The main bottleneck in compiling systems is **the CPU\_Main memory speed gap**.

**True/False Question:**

- 1)   F   Languages with structures are less readable than ones with parallel arrays.
- 2)   F   Most scripting languages are implemented using compiling systems.
- 3)   T   A language based on variables, assignment statement, and iteration is imperative.
- 4)   T   Reducing the feature multiplicity improves the language readability.
- 5)   T   APL language is better writeable than C++ language in arrays processing.

Student Name:

Student id:

Sect #: Serial #:

University of Bahrain

College of Information Technology  
Department of Computer Science

**ITCS332: Concepts of Programming Languages**

**QUIZ#2: Chapter 3\_Syntax**

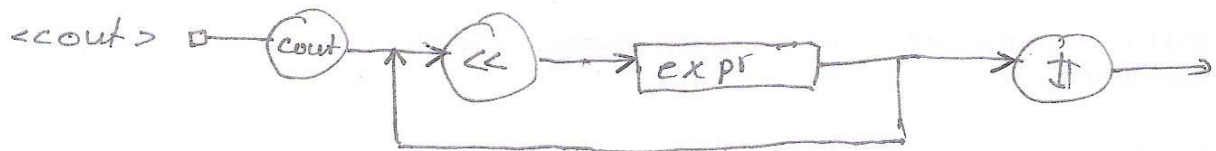
\*\*\*\*\*

**QUESTION 1:** Construct the BNF rules to define the C++ **cout** statement. A **cout** statement is a keyword **cout** followed by the insertion operator **<<** followed by one or more expressions separated by **<<** and terminated by **#**. Samples of **cout** statements: **cout << <uuu>#**  
**cout << <ex> << <tt>#**

3

**<cout>**       $\rightarrow$  **cout << <exprs> #**  
**<exprs>**     $\rightarrow$  **<expr> | <expr> << <exprs>**

**QUESTION 2:** Construct the syntax graphs of all nonterminals used in question #1.



**QUESTION 3:** Given the following BNF rules

**<digit> ::= 0|1|2|3|4|5|6|7|8|9.**

Construct BNF rules that define a hexadecimal integer **<HEX>** with no leading letters. A hexadecimal integer contains decimal digits and letters from **A (a)** to **F (f)**, terminates with **H (h)**. Examples of accepted values: **27H, 459h, 9A7Fh, 0Ah**

Examples of rejected values: **C27, 24, D009h, 2A7TH, 5B 67H**

4

**<HEX>**       $\rightarrow$  **<SEQ> H | <SEQ> h**  
**<SEQ>**       $\rightarrow$  **<DEC> | <SEQ> <digit> | <SEQ> <LET>**  
**<LET>**       $\rightarrow$  **A|B|C|D|E|F|a|b|c|d|e|f**



Student Name:

Student id:

Sect #: Serial #:

University of Bahrain

Department of Computer Science

College of Information Technology

ITCS332: Concepts of Programming Languages

QUIZ#2: Chapter 3\_Syntax

\*\*\*\*\*

QUESTION 1: Given the following BNF rules

<letter> ::= A|B|C|...|Z|a|b|c|...|z

<digit> ::= 0|1|2|3|4|5|6|7|8|9.

Construct the BNF rules to define the C++ **cin** statement defined as follows. A **cin** statement is a keyword **cin** followed by the extraction operator **>>** followed by one or more variables separated by **>>** and terminated by a semicolon.

Samples of **cin** statements: **cin >> test; cin >> t >> row4; cin >> d >> h >> b49;**

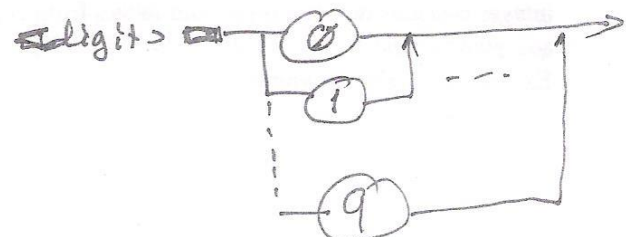
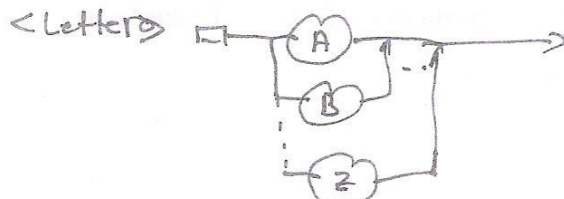
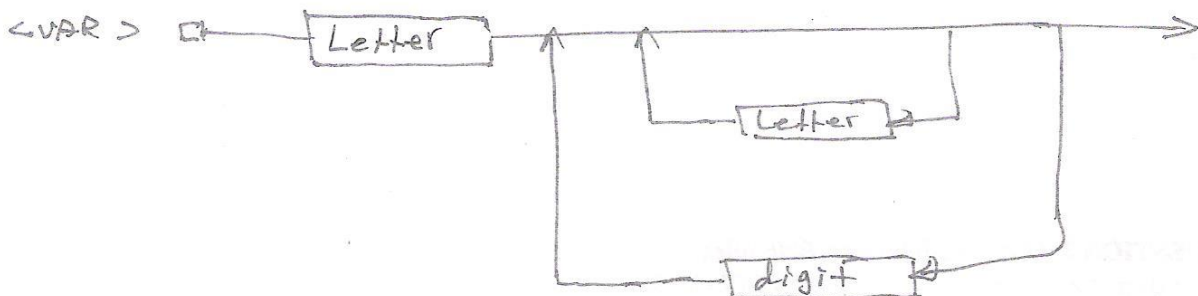
<cin> → cin >> <vars> ;

<vars> → <var> | <var> >> <vars>

QUESTION 2: Construct the BNF rules of a variable defined as a letter followed by zero or more digits/letters.

<var> → <letter>  
| <var><letter>  
| <var> <digit>

QUESTION 3: Construct the syntax graphs of all nonterminals used in question #2.



Student Name:

Stud id: 2013

Sect#: 2 #: 24

University of Bahrain

College of Information Technology  
Department of Computer Science

ITCS332: Concepts of Programming Languages

QUIZ#4: Chapter 16\_LP theory

\*\*\*\*\*

- 1) Prolog operates in 2 modes: rules or query.
- 2) The process of finding a complete sequence of propositions (proof) for the first subgoal before working on others is called depth-first searching; the process that works on all subgoals of a given goal in parallel is called breadth-first searching.
- 3) In Prolog, control enters a goal through its call port or through its redo port.
- 4) Name 2 kinds of Prolog statements: rules statement and facts statement.
- 5) In prolog, computations are performed by unification operator and unification is done by operator.
- 6) A clausal form of propositions contains antecedent operators in its left side and consequent operators in its right side.
- 7) The left side is called consequent and the right side of a rule is called antecedent.
- 8) The prolog query:  $[[t, b, 8] | X] = [Y, 9, [6, a, f], abc]$  . produces:  
 $Y = [t, b, 8]$  and  $X = [9, [6, a, f], abc]$
- 9) The prolog query:  $?-[X|Y] = [[isa, nooh], hud, [[ahmed], mustafa]]$  . produces:  
 $X = [isa, nooh]$  and  $Y = [ahmed, mustafa]$
- 10) The prolog query:  $?-Feb = feb, [B, Feb | A] = [[nov, 17, 2013], Feb, [Feb, 20, 2012]]$  . produces:  
 $B = nov$  and  $A = [Feb, 20, 2012]$
- 11) The prolog query:  $?- X = 20, X \text{ is } 10 * 2$  . produces:  $X = 20$  yes

Student Name:

Stud id: 2010

Sect#: 2 #: 24

University of Bahrain

College of Information Technology  
Department of Computer Science

ITCS332: Concepts of Programming Languages

Quiz #5: Chapter 16\_Prolog coding

\*\*\*\*\*

**QUESTION:** Define a prolog predicate `mySq` that displays the square of the last item in a given list `L` of numbers.

**For example,**

?- mySq([8,5,12],F).  
F = 144

?- mySq([1,2,-9,7,9,-8],M).  
M = 64

?- mySq([9],Q).  
Q = 81

?- mySq([],U).  
U = 0

**ANSWER:**

mySq([],0).  
mySq([\_],X):- X is A<sup>2</sup>.  
mySq([\_],X):- mySq([],X).



Student Name:

Stud id:

sect #: serial#:

University of Bahrain

College of Information Technology  
Department of Computer Science

ITCS332: Concepts of Programming Languages

Quiz#6: Chapter 5\_Names

\*\*\*\*\*

**QUESTION ONE:** What will be printed after executing the following C++ code? [6 pts]

```
void main()
{ int x = 12, y = 5, f = x+y;
  { x--; f = x - y;
    cout << x << '\t' << f << endl;
    { int x = 8, y = 9, f = x+y;
      cout << f << endl;
    }
    x--; y++; f = x * y;
    cout << x << '\t' << f << endl;
  }
}
```

11	6
17	
10	60

11	6
17	
10	60

**QUESTION TWO:** Fill in blanks

[5 pts]

- 6) With static type binding languages (such as FORTRAN), the type of a variable is specified using Explicit declaration statement or Explicit declaration.
- 7) The lifetime of a stack-dynamic variable begins whenever the defining program unit/ block is entered and ends at whenever the defining program unit/block is exited.
- 8) Stack-dynamic variables are allocated storage when the defining program unit/ block is entered and deallocated when it is exited. All attributes of implicit-heap dynamic variables are bound every time they are assigned values.
- 9) Name 2 ways used to create aliases in C++: pointers (references), and function parameters.
- 10) A call to a library function is bound to the function code in the library at link time, while a data type such as double is bound to a range of possible values at language implementation time.

Student Name:

Stud id:

sect #: serial#:

University of Bahrain

College of Information Technology  
Department of Computer Science

ITCS332: Concepts of Programming Languages

Quiz#6: Chapter 5\_Names

\*\*\*\*\*

QUESTION ONE: Fill in blanks

[5 pts]

- 1) The “\*” symbol is bound to “multiplication” operation at **language design** time, and a call to a library function “pow” is bound to the function code in the library at **link** time.
- 2) The lifetime of a static variable begins when **the defining program unit/ block is FIRSTLY loaded** and ends when **the entire program terminates**.
- 3) With dynamic type binding used in JavaScript and PHP, the type of a variable is specified using **referencing context** or **whenever a referencing Assignment statement is executed**.
- 4) In a dynamic-scoped language, the referencing environment depends on **the calling sequence** of program units. In a static-scoped language; it depends on **the textual layout** of program unit/blocks.
- 5) For every assignment to a subrange variable, the **Range checking** is done at run-time, and the **Type checking** is done at compile time.

QUESTION TWO: What will be printed after executing the following C++ code?

[6 pts]

```
void main()
{ int x = 8, y = 11, f = x+y;
  { x--; f = x - y;
    cout << x << '\t' << f << endl;
    { int x = 5, y = 8, f = x+y;
      cout << f << endl;
    }
    x--; y++; f = x * y;
    cout << x << '\t' << f << endl;
  }
}
```

7	-4
13	
6	72